

[illegible][illegible]

```
SSSSSSSS  UU      UU  MM      MM  MM      MM      AAAAAA  IIIIII  NN      NN
SSSSSSSS  UU      UU  MM      MM  MM      MM      AAAAAA  IIIIII  NN      NN
SS      UU      UU  MMMM  MMMM  MMMM  MMMM  AA      AA  II      NN      NN
SS      UU      UU  MMMM  MMMM  MMMM  MMMM  AA      AA  II      NN      NN
SS      UU      UU  MM  MM  MM  MM  MM  AA      AA  II      NNNN  NN
SSSSSS  UU      UU  MM      MM  MM      MM  AA      AA  II      NNNN  NN
SSSSSS  UU      UU  MM      MM  MM      MM  AA      AA  II      NN  NN  NN
SS      UU      UU  MM      MM  MM      MM  AAAAAAAAAA  II      NN      NNNN
SS      UU      UU  MM      MM  MM      MM  AAAAAAAAAA  II      NN      NNNN
SS      UU      UU  MM      MM  MM      MM  AA      AA  II      NN      NN
SSSSSSSS  UUUUUUUUUU  MM      MM  MM      MM  AA      AA  IIIIII  NN      NN
SSSSSSSS  UUUUUUUUUU  MM      MM  MM      MM  AA      AA  IIIIII  NN      NN
                                     ....
                                     ....
                                     ....
                                     ....
```

```
LL      IIIIII  SSSSSSSS
LL      IIIIII  SSSSSSSS
LL      II      SS
LL      II      SS
LL      II      SS
LL      II      SSSSSS
LL      II      SSSSSS
LL      II      SS
LL      II      SS
LL      II      SS
LL      II      SS
LLLLLLLLLL  IIIIII  SSSSSSSS
LLLLLLLLLL  IIIIII  SSSSSSSS
```



(5) 289

RETURN\_NODES



```
0000 1 :
0000 2 : Version: 'V04-000'
0000 3 :
0000 4 : *****
0000 5 : *
0000 6 : * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *
0000 7 : * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *
0000 8 : * ALL RIGHTS RESERVED. *
0000 9 : *
0000 10 : * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
0000 11 : * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *
0000 12 : * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
0000 13 : * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
0000 14 : * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
0000 15 : * TRANSFERRED. *
0000 16 : *
0000 17 : * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
0000 18 : * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
0000 19 : * CORPORATION. *
0000 20 : *
0000 21 : * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
0000 22 : * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *
0000 23 : *
0000 24 : *
0000 25 : *****
0000 26 :
0000 27 :
0000 28 : Assembly parameters
0000 29 :
00000200 0000 30 : BUF_SIZE = 512 ; Size in bytes of slipr input buffers
00000084 0000 31 : CMD_SIZE = 132 ; Size of input command line
0000 32 :
0000 33 : $NAMDEF
0000 34 : $RABDEF
0000 35 : $FABDEF
0000 36 : $CLIDEF
0000 37 :
0000 38 : Edit node offsets
0000 39 :
00000000 0000 40 : ED$L_FWD = 0 ; Forward pointer
00000004 0000 41 : ED$B_BWD = 4 ; Backward pointer
00000008 0000 42 : ED$W_LOC1 = 8 ; Locator 1
0000000A 0000 43 : ED$W_LOC2 = 10 ; Locator 2
0000000C 0000 44 : ED$W_LINES = 12 ; Insert lines
0000000E 0000 45 : ED$W_RFA = 14 ; Record file address (3 words)
00000014 0000 46 : ED$B_FILE = 20 ; File node pointer
00000018 0000 47 : ED$B_FLAGS = 24 ; Flags
00000019 0000 48 : ED$B_FILENO = 25 ; File number
0000 49 :
0000001A 0000 50 : ED$K_BLN = 26
0000 51 :
0000 52 :
0000 53 : File node offsets
0000 54 :
00000000 0000 55 : SLP$L_FWD = 0 ; Forward pointer
00000004 0000 56 : SLP$B_BWD = 4 ; Backward pointer
00000008 0000 57 : SLP$W_LOC1 = 8 ; Locator-1
```



```

0000000A 0000 58      SLP$W_LOC2 = 10      ; Locator-2
0000000C 0000 59      SLP$B_FLAGS= 12      ; Flags
0000000D 0000 60      SLP$B_FILENO = 13      ; File priority
0000000E 0000 61      SLP$W_DOT = 14      ; Dot value
00000010 0000 62      SLP$Q_AUDDS= 16      ; Audit string descriptor
00000018 0000 63      SLP$T_AUDST= 24      ; Audit string
00000028 0000 64      SLP$Q_AUCDS= 40      ; Current audit string descriptor
00000030 0000 65      SLP$T_AUCST= 48      ; Current audit string
00000040 0000 66      SLP$Q_CMNT = 64      ; Comment descriptor
00000048 0000 67      SLP$T_NAM = 72      ; NAM block
          0000 68      ;
000000A8 0000 69      SLP$K_BLN = SLP$T_NAM + NAM$K_BLN
          0000 70      ;
          0000 71      ;
          0000 72      ; Macro to print error message
          0000 73      ;
          0000 74      .MACRO ERRMSG NAME,LIST
          0000 75      $$ = 0
          0000 76      .IRP L,<LIST>
          0000 77      PUSHL L
          0000 78      $$=$$+1
          0000 79      .ENDR
          0000 80      PUSHL #$$
          0000 81      MOVL #MERS_'NAME',R0
          0000 82      PUSHL R0
          0000 83      CALLS #$$+2,G^LIB$SIGNAL
          0000 84      .ENDM ERRMSG

```

SUM  
Sym[illegible]



```
0000 1 .TITLE SUM -- SOURCE UPDATE MERGE
0000 2 .IDENT /V04-000/
0000 3
0000 4 *****
0000 5
0000 6 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 7 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 8 * ALL RIGHTS RESERVED.
0000 9
0000 10 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 11 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 12 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 13 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 14 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 15 * TRANSFERRED.
0000 16
0000 17 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 18 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 19 * CORPORATION.
0000 20
0000 21 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 22 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 23
0000 24 *****
0000 25 *****
0000 26
0000 27
0000 28
0000 29 System equated symbols
0000 30
0000 31 $CHFDEF ; Condition handling defintions
0000 32 $STSDEF ; Condition handling fields
0000 33 $CLIVERBDEF ; Generic verb types
0000 34
0000 35 .PSECT $DATA,NOEXE,WRT
0000 36
0000 37 Command input
0000 38
0000 39 CMD_INPUT FAB:
0000 40 $FAB FNM = <SYSS$INPUT> - ; File name
0000 41 FOP = INP - ; Process permanent file
0000 42 FAC = GET ; Allow GET$ operations
0000 43
0000 44 CMD_INPUT RAB::
0000 45 $RAB FAB = CMD_INPUT_FAB - ; File access block address
0000 46 UBF = CMD_INPUT_BUF - ; Command input buffer
0000 47 USZ = CMD_SIZE = ; Command input buffer size
0000 48 ROP = <PMT,CVT>
0000 49 ; Convert lower case to upper case
0000 50 ; Read with prompt
0000 51
0000 52
0000 53 Command output
0000 54
0000 55 CMD_OUTPUT FAB:
0000 56 $FAB FNM = <SYSS$OUTPUT> - ; File name
0000 57 FAC = PUT - ; Allow PUT$ operations
```



```
0094 58          RAT = CR          ; Lines require CR/LF
00E4 59 :
00E4 60 CMD_OUTPUT_RAB:
00E4 61          $RAB      FAB = CMD_OUTPUT_FAB, - ; File access block address
00E4 62          RBF = CMD_OUTPUT_BUF ; Record address
0128 63 :
0128 64 :
0128 65 : Prompt strings
0128 66 :
0128 67 PROMPT_INPUT::
65 6C 69 66 20 74 75 70 6E 49 5F 00' 0128 68          .ASCIC  "_Input files:  "
20 20 3A 73 0134
OF 0128
0138 69 :
0138 70 PROMPT_OUTPUT::
6C 69 66 20 74 75 70 74 75 4F 5F 00' 0138 71          .ASCIC  "_Output file:  "
20 20 3A 65 0144
OF 0138
0148 72 :
0148 73 PROMPT_CONT::
20 5F 00' 0148 74          .ASCIC  "_ "
02 0148
014B 75 :
014B 76 :
014B 77 : Command input buffer
014B 78 :
014B 79 CMD_INPUT_BUF:
000001CF 014B 80          .BLKB  CMD_SIZE
01CF 81 :
00000000 01CF 82 CMD_INPUT_SIZE:: .LONG 0          ; Current string size
00000000 01D3 83 CMD_INPUT_POS:: .LONG 0          ; Current scan position
01D7 84 :
01D7 85 REQ_COMMAND::
01D7 86          $CLIREQDESC RQTYPE = CLISK_GETCMD
01F3 87 :
01F3 88 :
00000084 01F3 89 CMD_OUTPUT_DESC: .LONG  CMD_SIZE
000001FB' 01F7 90          .LONG  CMD_OUTPUT_BUF
0000027F 01FB 91 CMD_OUTPUT_BUF: .BLKB  CMD_SIZE
027F 92 :
027F 93 :
00000281 027F 94 SLP_FLAGS:: .BLKW 1
0281 95 :
00000289'00000010 0281 96 SLP_AUDDS:: .LONG 16,SLP_AUDST
0289 97 :
00000299 0289 98 SLP_AUDST:: .BLKB 16
0299 99 :
0299 100 :
0000 0299 101 MERGE_FLAGS:: .WORD 0
029B 102 :
029B 103 : Flag bits
029B 104 :
00000001 029B 105          MERM_OUTPUT == 1          ; Output file specified
029B 106          ; 0 = No 1 = Yes
029B 107 :
029B 108 :
029B 109 : List heads
```



```
0000029B'0000029B' 029B 110 :  
029B 111 EDIT_NODES::  
029B 112 .LONG EDIT_NODES,EDIT_NODES  
02A3 113 :  
000002A3'000002A3' 02A3 114 FILE_NODES::  
02A3 115 .LONG FILE_NODES,FILE_NODES  
02AB 116 :  
00000000 02AB 117 VIRT_ADDR::  
02AB 118 .LONG 0  
02AF 119 :  
000000A8 02AF 120 SLP_SIZE::  
02AF 121 .LONG SLP$K_BLN  
02B3 122 :  
0000001A 02B3 123 ED_SIZE::  
02B3 124 .LONG ED$K_BLN  
02B7 125 :  
00000000 02B7 126 FILE_SIZE::  
02B7 127 .LONG 0  
02BB 128 :  
02BB 129 ; Strings used to generate edit commands  
02BB 130 :  
25 02BB 131 AUDIT_ON:: .ASCII "%"  
5C 02BC 132 AUDIT_OFF:: .ASCII "\"  
2F 02BD 133 END_EDIT:: .ASCII "/"  
57 55 21 2D 000002C6'010E0000' 02BE 134 LOC_ONE:: .ASCID "-!UW"  
57 55 21 2C 000002D2'010E0000' 02CA 135 LOC_TWO:: .ASCID "-!UW"  
2C 000002DE'010E0000' 02D6 136 COMMAS:: .ASCID ",,"  
2F 53 41 21 2F 2C 000002E7'010E0000' 02DF 137 AUDIT_TRAIL:: .ASCID " /!AS/"  
53 41 21 3B 000002F5'010E0000' 02ED 138 COMMENTS:: .ASCID " :!AS"  
02F9 139 :  
00000301 02F9 140 OUTDES:: .BLKL 2  
00000305 0301 141 OUTLEN:: .BLKL 1  
0305 142 :  
0305 143 :  
44 50 55 2E 0000030D'010E0000' 0305 144 DEF_NAME:: .ASCID /.UPD/  
0311 145 :  
57 45 4E 2A 2A 3B 00000319'010E0000' 0311 146 DEF_AUDIT:: .ASCID /:**NEW**/  
2A 2A 031F
```



```
0321 148 :  
0321 149 : RMS blocks  
0321 150 :  
0321 151 : Flag bits  
0321 152 :  
00000001 0321 153 : SLPM_OPEN == 1 : File open flag  
00000002 0321 154 : SLPM_AUDIT == 2 : Audit trail on/off  
00000004 0321 155 : SLPM_SUPPRESS == 4 : Suppress clash report  
00000002 0321 156 : SLPV_SUPPRESS == 2 : ...  
0321 157 :  
0321 158 :  
0321 159 : Input and Output file blocks  
0321 160 :  
0321 161 :  
00000000 0321 162 : .SAVE  
0000 163 : .PSECT RMS  
0000 164 OUTPUT_FAB::  
0000 165 $FAB FAC = PUT -  
0000 166 RAT = CR -  
0000 167 NAM = OUTPUT_NAM -  
0000 168 DNM = <.UPD>  
0050 168 :  
0050 169 OUTPUT_RAB::  
0050 170 $RAB FAB = OUTPUT_FAB -  
0050 171 RAC = SEQ -  
0050 172 UBF = OUTPUT_BUF -  
0050 173 USZ = BUF_SIZE  
0094 174 :  
00000294 0094 175 OUTPUT_BUF::  
0094 176 .BLKB BUF_SIZE  
0294 177 :  
0294 178 OUTPUT_NAM::  
0294 179 $NAM ESA = OUTPUT_BUF -  
0294 180 ESS = 255  
02F4 181 :  
02F4 182 :  
02F4 183 INPUT_FAB::  
02F4 184 $FAB FAC = GET  
0344 185 :  
0344 186 INPUT_RAB::  
0344 187 $RAB FAB = INPUT_FAB -  
0344 188 RAC = SEQ -  
0344 189 UBF = INPUT_BUF -  
0344 190 USZ = BUF_SIZE  
0388 191 :  
00000588 0388 192 INPUT_BUF::  
0388 193 .BLKB BUF_SIZE  
0588 194 :  
0588 195 :  
0588 196 RANDOM_FAB::  
0588 197 $FAB FAC = GET -  
0588 198 FOP = NAM  
05D8 199 :  
05D8 200 RANDOM_RAB::  
05D8 201 $RAB FAB = RANDOM_FAB -  
05D8 202 RAC = RFA -  
05D8 203 UBF = RANDOM_BUF -  
05D8 204 USZ = BUF_SIZE
```



SUM  
V04-000

-- SOURCE UPDATE MERGE

J 8

16-SEP-1984 02:14:36 VAX/VMS Macro V04-00  
5-SEP-1984 16:56:59 [SUM.SRC]SUMMAIN.MAR;1

Page 7  
(2)

```
0000081C 061C 205 ;
061C 206 RANDOM_BUF::
061C 207 .BLKB BUF_SIZE
081C 208 ;
081C 209 RANDOM_FILE::
0000081C 081C 210 .BLKL 0
081C 211 ;
081C 212 ;
00000321 213 .RESTORE
```

\*\*\*F



```
0321 215 ;
00000000 216 .PSECT $CODE,NOWRT,EXE
0000 217
0000 218 START::
0000 219
6D 017D'CF 0000 220 .WORD 0
DE 0002 221 MOVAL W^HANDLER,(FP) ; Set handler address
00000085'EF 00 FB 0007 222 ;
70 50 E9 000E 223 CALLS #0,INITIALISE ; Perform initialisation
0000 224 10$: BLBC R0,100$ ; Error if LBC
7E 7C 0011 225 CLRQ -(SP) ; Get any command line from DCL
01D7'CF DF 0013 226 PUSHAL W^REQ_COMMAND
08 BC 03 FB 0017 227 CALLS #3,ACCLISA_UTILSERV(AP)
FF 8F 000001DA'EF 91 001B 228 CMPB REQ_COMMAND+CLISB_RQSTAT, - ; Was program invoked as a foreign
0000 229 #CLISK_VERB_FORE ; command?
14 12 0023 230 BNEQ 20$ ; No if NEQ
01CF'CF 000001DF'EF 3C 0025 231 MOVZWL REQ_COMMAND+CLISW_RQSIZE, -
0000 232 W^CMD_INPUT_SIZE
01D3'CF 000001E3'EF D0 002E 233 MOVL REQ_COMMAND+CLISA_RQADDR, -
0000 234 W^CMD_INPUT_POS
04 11 0037 235 BRB 30$
0000 236 20$:
01CF'CF D4 0039 237 CLRL W^CMD_INPUT_SIZE ; Initialise input buffer
0000 238 30$:
50 01 D0 003D 239 MOVL #1,R0
00000000'EF 00 FB 0040 240 CALLS #0,GET_FILES ; Get input and output files
0B 50 E8 0047 241 BLBS R0,40$ ; OK if LBS
00000000'8F 50 D1 004A 242 CMPL R0,#RMS$_EOF ; Was error EOF?
1A 12 0051 243 BNEQ 60$ ; No if NEQ
2C 11 0053 244 BRB 100$
0000 245 40$:
00000000'EF 00 FB 0055 246 CALLS #0,OPEN_FILES ; Open files
07 50 E9 005C 247 BLBC R0,50$ ; Error if LBC
0000 248 ;
00000000'EF 00 FB 005F 249 CALLS #0,MERGE_FILES ; Merge files
0000 250 50$:
00000000'EF 00 FB 0066 251 CALLS #0,CLOSE_FILES ; Close files
FF 8F 000001DA'EF 91 006D 252 60$:
0000 253 CMPB REQ_COMMAND+CLISB_RQSTAT, - ; Was program invoked as a foreign
0000 254 #CLISK_VERB_FORE ; command?
0A 13 0075 255 BEQL 100$ ; Yes if EQL
000000DF'EF 00 FB 0077 256 CALLS #0,RETURN_NODES
B8 50 E8 007E 257 BLBS R0,20$ ; OK if LBS
50 01 D0 0081 258 100$:
0000 259 MOVL #1,R0
04 0084 260 RET
0085 261
```



```
0085 263 :  
0085 264 : Procedure to perform initialisation  
0085 265 :  
0085 266 :  
0085 267 INITIALISE::  
0000 0085 268 .WORD 0  
10 50 E9 0087 269 $OPEN FAB = CMD_INPUT_FAB ; Open command input  
08 50 E8 0094 270 BLBC R0,10$ ; Error if LBC  
0097 271 $CONNECT RAB= CMD_INPUT_RAB ; Connect to RAB  
00A4 272 BLBS R0,20$ ; OK if LBS  
00A7 273 10$:  
0000000C'EF DD 00A7 274 PUSHL CMD_INPUT_FAB+FAB$L_STV ; Push error status  
26 11 00AD 275 BRB 40$  
00AF 276 20$:  
10 50 E9 00AF 277 $OPEN FAB = CMD_OUTPUT_FAB ; Open command output  
00BC 278 BLBC R0,30$ ; Error if LBC  
00BF 279 $CONNECT RAB= CMD_OUTPUT_RAB ; Connect to RAB  
0F 50 E8 00CC 280 BLBS R0,50$ ; OK if LBS  
00CF 281 30$:  
000000A0'EF DD 00CF 282 PUSHL CMD_OUTPUT_FAB+FAB$L_STV; Push error status  
50 DD 00D5 283 40$:  
00000000'GF 02 FB 00D5 284 PUSHL R0  
00D7 285 CALLS #2,G^LIB$SIGNAL ; Signal error  
00DE 286 50$:  
04 00DE 287 RET
```



```
00DF 289 .SBTTL RETURN_NODES
00DF 290 :
00DF 291 : Procedure to return all nodes to virtual memory
00DF 292 :
00DF 293 : Inputs:
00DF 294 : None
00DF 295 :
00DF 296 : Outputs:
00DF 297 : R0 = Success/error status
00DF 298 :
00DF 299 :
00DF 300 RETURN_NODES:
0000 00DF 301 .WORD 0
00E1 302 10$:
55 029B'DF 0F 00E1 303 REMQUE @W^EDIT_NODES,R5 ; Get edit node
19 1D 00E6 304 BVS 20$ ; Queue now empty if VS
02AB'CF 55 D0 00E8 305 MOVL R5,W^VIRT_ADDR ; Store size
02AB'CF DF 00ED 306 PUSHAL W^VIRT_ADDR ; Push parameters
02B3'CF DF 00F1 307 PUSHAL W^ED_SIZE
00000000'GF 02 FB 00F5 308 CALLS #2,G^LIB$FREE_VM
61 50 E9 00FC 309 BLBC R0,50$
E0 11 00FF 310 BRB 10$
55 02A3'CF DE 0101 311 20$:
0106 312 MOVAL W^FILE_NODES,R5 ; Get file nodes list head
55 65 D0 0106 313 30$:
000002A3'8F 55 D1 0109 314 MOVL (R5),R5 ; Get next NODE
2E 13 0110 315 CMPL R5,#FILE_NODES ; At end of list?
54 55 D0 0112 316 BEQL 40$ ; Yes if EQL
54 00000048 8F C0 0115 317 MOVL R5,R4 ; Set NAM pointer
02AB'CF 04 A4 D0 011C 318 ADDL2 #SLP$T NAM,R4
E2 13 0122 319 MOVL NAM$SL_RSA(R4),W^VIRT_ADDR ; Get file name node address
02B7'CF 03 A4 9A 0124 320 BEQL 30$
DA 13 012A 321 MOVZBL NAM$B_RSL(R4),W^FILE_SIZE ; and size
02AB'CF DF 012C 322 BEQL 30$
02B7'CF DF 0130 323 PUSHAL W^VIRT_ADDR ; Push parameters
00000000'GF 02 FB 0134 324 PUSHAL W^FILE_SIZE
22 50 E9 0138 325 CALLS #2,G^LIB$FREE_VM
C6 11 013E 326 BLBC R0,50$
0140 327 BRB 30$
55 02A3'DF 0F 0140 328 40$:
19 1D 0145 329 REMQUE @W^FILE_NODES,R5 ; Get file node
02AB'CF 55 D0 0147 330 BVS 50$ ; Queue now empty if VS
02AB'CF DF 014C 331 MOVL R5,W^VIRT_ADDR ; Store size
02AF'CF DF 0150 332 PUSHAL W^VIRT_ADDR ; Push parameters
00000000'GF 02 FB 0154 333 PUSHAL W^SLP_SIZE
02 50 E9 0158 334 CALLS #2,G^LIB$FREE_VM
E0 11 015E 335 BLBC R0,50$
04 0160 336 BRB 40$
0160 337 50$:
0160 338 RET
```

```
0161 340 :  
0161 341 : Condition handlers  
0161 342 :  
0161 343 : .ENABL LSB  
0161 344 :  
0161 345 GET_HANDLER::  
0161 346 .WORD ^M<R2,R3,R4,R5>  
0163 347 MOVL CHF$SIGARGLST(AP),R2 ; Get address of signal array  
0167 348 CMPL CHF$SIG_NAME(R2),#RMS$_EOF  
016F 349 BNEQ 10$  
0171 350 $UNWIND_S  
017C 351 RET  
017D 352 :  
017D 353 :  
017D 354 HANDLER::  
017D 355 .WORD ^M<R2,R3,R4,R5>  
017F 356 10$:  
017F 357 MOVL CHF$SIGARGLST(AP),R2 ; Get address of signal array  
0183 358 MOVL CHF$SIG_NAME(R2),R0 ; Get message name  
0187 359 BITL #STSS$_FAC_NO,R0 ; Is facility number empty?  
018E 360 BNEQ 99$ ; No if NEQ  
0190 361 BBC #STSS$_FAC_SP,R0,99$ ; Facility specific bit must be set  
0194 362 MOVAL W^MERGE_MSGS,R1 ; Get address of Merge messages  
0199 363 20$:  
0199 364 MOVQ (R1)+,R4 ; Get next entry  
019C 365 BEQL 30$ ; End of table if EQL  
019E 366 CMPL R0,R4 ; This message?  
01A1 367 BNEQ 20$ ; No if NEQ  
01A3 368 BRB 40$  
01A5 369 30$:  
01A5 370 MOVAL W^CATCH_MSG,R5 ; Use catch message  
01AA 371 40$:  
01AA 372 $FAOL_S CTRSTR = (R5), - ; Control string  
01AA 373 OUTLEN = CMD_OUTPUT_RAB+RAB$_RSZ, -  
01AA 374 OUTBUF = CMD_OUTPUT_DESC, -  
01AA 375 PRMLST = 12(R2)  
01C2 376 $PUT RAB = CMD_OUTPUT_RAB  
01CF 377 EXTZV #STSS$_SEVERITY,#STSS$_SEVERITY, -  
01D5 378 CHF$SIG_NAME(R2),R0 ; Extract severity code  
01D5 379 CMPL R0,#STSS$_WARNING ; Is it a warning?  
01D8 380 BNEQ 99$ ; No if NEQ  
01DA 381 MOVL #SS$_CONTINUE,R0  
01E1 382 RET  
01E2 383 99$:  
01E2 384 MOVL #SS$_RESIGNAL,R0  
01E9 385 RET  
01EA 386 ;  
01EA 387 .DSABL LSB
```



```
01EA 389 : Macro to define error messages
01EA 390 :
01EA 391 :
01EA 392 :
01EA 393 .MACRO MSG NAME,FAOCNT,TEXT,?L1
01EA 394 .SAVE
01EA 395 .PSECT ERRORS
01EA 396 L1: .ASCII "TEXT"
01EA 397 .RESTORE
01EA 398 .LONG MERS_'NAME
01EA 399 .LONG L1
01EA 400 .ENDM MSG
01EA 401 :
01EA 402 MERGE_MSGS:
01EA 403 MSG PRSERR,2,<Error parsing file specification !AD>
01F2 404 MSG NULLFS,0,<Null file specification>
01FA 405 MSG ONEOUT,0,<Only one output file allowed>
0202 406 MSG OPENER,2,<Error opening !AD for input>
020A 407 MSG CREATE,2,<Error creating !AD for output>
0212 408 MSG OUTERR,0,<Error writing to output file>
021A 409 MSG INPEOF,1,<Unexpected EOF on input file !AD>
0222 410 MSG CLASH,0,<Edit clashes....>
022A 411 MSG CLSHLN,4,< File !AD!/ !AD>
0232 412 MSG NONODE,0,<Can't expand program>
023A 413 MSG ATSIGN,2,<a command illegal!/!AD>
0242 414 MSG REOPEN,2,<Error re-opening !AD for input>
024A 415 MSG CONNEX,2,<Error connecting !AD to RAB>
0252 416 MSG DISCON,2,<Error disconnecting !AD from RAB>
025A 417 MSG CLOSER,2,<Error closing !AD>
0262 418 MSG READER,2,<Error reading file !AD>
026A 419 MSG INVPMO,0,<Invalid parameter delimiter>
0272 420 MSG SYNTAX,2,<SLP command syntax error!/ !AD>
00000000 00000000 027A 421 .QUAD 0
0282 422 :
0282 423 CATCH_MSG:
0282 424 MSG CTCHER,0,<Merge condition handler error>
028A 425 :
028A 426 :
028A 427 : Define message name symbols
028A 428 :
028A 429 $EQLST MERS_,GLOBAL,32778,8,< -
028A 430 <CTCHER> -
028A 431 <NONODE> -
028A 432 >
028A 433 :
028A 434 :
028A 435 $EQLST MERS_,GLOBAL,34824,8,< -
028A 436 <NULLFS> -
028A 437 <ONEOUT> -
028A 438 <PRSERR> -
028A 439 <OPENER> -
028A 440 <CREATE> -
028A 441 <OUTERR> -
028A 442 <INPEOF> -
028A 443 <CLASH> -
028A 444 <CLSHLN> -
028A 445 <ATSIGN> -
```



028A 446  
028A 447  
028A 448  
028A 449  
028A 450  
028A 451  
028A 452  
028A 453  
028A 454 :  
028A 455 :  
028A 456 :

<REOPEN> -  
<CONNEC> -  
<DISCON> -  
<CLOSER> -  
<READER> -  
<INVPMD> -  
<SYNTAX> -  
>

.END START



Variable	Value	Mode	Size
\$\$TAB	= 000005D8	R	04
\$\$TABEND	= 0000061C	R	04
\$\$TMP	= 00000000		
\$\$TMP1	= 00000001		
\$\$TMP2	= 000000CF		
\$\$TMPX	= 00000013	R	03
\$\$TMPX1	= 00000004		
\$CLI.	= 000001D7	R	02
\$CLI..	= 000001F3	R	02
AUDIT_OFF	000002BC	RG	02
AUDIT_ON	000002BB	RG	02
AUDIT_TRAIL	000002DF	RG	02
BIT...	= 00008890		
BUF_SIZE	= 00000200		
CATCH_MSG	00000282	R	05
CHFSL_SIGARGLST	= 00000004		
CHFSL_SIG_NAME	= 00000004		
CLISA_RQADDR	= 0000000C		
CLISA_UTILSERV	= 00000008		
CLISB_RQSTAT	= 00000003		
CLISB_RQTYPE	= 00000000		
CLISC_REQDESC	= 0000001C		
CLISK_GETCMD	= 00000001		
CLISK_VERB_FORE	= 000000FF		
CLISW_RQSIZE	= 00000008		
CLOSE_FILES	*****	X	05
CMD_INPUT_BUF	0000014B	R	02
CMD_INPUT_FAB	00000000	R	02
CMD_INPUT_POS	000001D3	RG	02
CMD_INPUT_RAB	00000050	RG	02
CMD_INPUT_SIZE	000001CF	RG	02
CMD_OUTPUT_BUF	000001FB	R	02
CMD_OUTPUT_DESC	000001F3	R	02
CMD_OUTPUT_FAB	00000094	R	02
CMD_OUTPUT_RAB	000000E4	R	02
CMD_SIZE	= 00000084		
COMMAS	000002D6	RG	02
COMMENTS	000002ED	RG	02
DEF_AUDIT	00000311	RG	02
DEF_NAME	00000305	RG	02
ED\$B_FILENO	= 00000019		
ED\$B_FLAGS	= 00000018		
ED\$K_BLN	= 0000001A		
ED\$L_BWD	= 00000004		
ED\$L_FILE	= 00000014		
ED\$L_FWD	= 00000000		
ED\$W_LINES	= 0000000C		
ED\$W_LOC1	= 00000008		
ED\$W_LOC2	= 0000000A		
ED\$W_RFA	= 0000000E		
EDIT_NODES	0000029B	RG	02
ED_SIZE	000002B3	RG	02
END_EDIT	000002BD	RG	02
FAB\$B_DNS	= 00000035		
FAB\$B_FNS	= 00000034		
FAB\$C_BID	= 00000003		
FAB\$C_BLN	= 00000050		

FAB\$C\_SEQ  
FAB\$C\_VAR  
FAB\$L\_ALQ  
FAB\$L\_DNA  
FAB\$L\_FNA  
FAB\$L\_FOP  
FAB\$L\_STV  
FAB\$V\_CHAN\_MODE  
FAB\$V\_CR  
FAB\$V\_FILE\_MODE  
FAB\$V\_GET  
FAB\$V\_INP  
FAB\$V\_LNM\_MODE  
FAB\$V\_NAM  
FAB\$V\_PUT  
FAB\$W\_GBC  
FILE\_NODES  
FILE\_SIZE  
GET\_FILES  
GET\_HANDLER  
HANDLER  
INITIALISE  
INPUT\_BUF  
INPUT\_FAB  
INPUT\_RAB  
LIB\$FREE\_VM  
LIB\$SIGNAL  
LOC\_ONE  
LOC\_TWO  
MERS\$\_AT\$IGN  
MERS\$\_CLASH  
MERS\$\_CLOSER  
MERS\$\_CL\$HLN  
MERS\$\_CONNEC  
MERS\$\_CREATE  
MERS\$\_CTCHER  
MERS\$\_DISCON  
MERS\$\_INPEOF  
MERS\$\_INVPMD  
MERS\$\_NONODE  
MERS\$\_NULLFS  
MERS\$\_ONEOUT  
MERS\$\_OPENER  
MERS\$\_OUTERR  
MERS\$\_PR\$ERR  
MERS\$\_READER  
MERS\$\_REOPEN  
MERS\$\_SYNTAX  
MERGE\_FILES  
MERGE\_FLAGS  
MERGE\_MSGS  
MERM\_OUTPUT  
NAM\$B\_\$\$S  
NAM\$B\_\$OP  
NAM\$B\_\$RSL  
NAM\$B\_\$RSS  
NAM\$C\_\$BID

```

= 00000000
= 00000002
= 00000010
= 00000030
= 0000002C
= 00000004
= 0000000C
= 00000002
= 00000001
= 00000004
= 00000001
= 00000013
= 00000000
= 00000018
= 00000000
= 00000048
000002A3 RG 02
000002B7 RG 02
***** X 05
00000161 RG 05
0000017D RG 05
00000085 RG 05
00000388 RG 04
000002F4 RG 04
00000344 RG 04
***** X 05
***** X 05
000002BE RG 02
000002CA RG 02
= 00008850 G
= 00008840 G
= 00008870 G
= 00008848 G
= 00008860 G
= 00008828 G
= 0000800A G
= 00008868 G
= 00008838 G
= 00008880 G
= 00008012 G
= 00008808 G
= 00008810 G
= 00008820 G
= 00008830 G
= 00008818 G
= 00008878 G
= 00008858 G
= 00008888 G
***** X 05
00000299 RG 02
000001EA R 05
= 00000001 G
= 0000000A
= 00000008
= 00000003
= 00000002
= 00000002

```

SS.  
SS.  
BIT  
FAB  
FAB  
FAB  
FAB  
FAB  
FAB  
LIB  
LIB  
NAM  
NAM  
NAM  
NAM  
NAM  
RE T  
RMS  
SIZ  
SUM  
SUM  
SUM  
SUM  
SUM  
SUM  
SUM  
SUM  
SUM  
SUM  
SUM  
SUM  
SUM  
SYS  
SYS  
UPF  
UPF  
UPF  
UPF  
UPF  
UPF  
UPF  
UPF  
UPF  
UPF



SUM  
Symbol table

-- SOURCE UPDATE MERGE

E 9

16-SEP-1984 02:14:36 VAX/VMS Macro V04-00  
5-SEP-1984 16:56:59 [SUM.SRC]SUMMAIN.MAR;1

Page 15  
(7)

```

NAMSC_BLN      = 00000060
NAMSK_BLN      = 00000060
NAMSL_ESA      = 0000000C
NAMSL_RSA      = 00000004
OPEN_FILES     ***** X 05
OUTDES         000002F9 RG 02
OUTLEN         00000301 RG 02
OUTPUT_BUF     00000094 RG 04
OUTPUT_FAB     00000000 RG 04
OUTPUT_NAM     00000294 RG 04
OUTPUT_RAB     00000050 RG 04
PROMPT_CONT    00000148 RG 02
PROMPT_INPUT   00000128 RG 02
PROMPT_OUTPUT  00000138 RG 02
RABSB_RAC      = 0000001E
RABSC_BID      = 00000001
RABSC_BLN      = 00000044
RABSC_RFA      = 00000002
RABSC_SEQ      = 00000000
RABSL_CTX      = 00000018
RABSL_ROP      = 00000004
RABSV_CVT      = 0000001A
RABSV_PMT      = 0000001E
RABSW_RSZ      = 00000022
RANDOM_BUF      0000061C RG 04
RANDOM_FAB      00000588 RG 04
RANDOM_FILE     0000081C RG 04
RANDOM_RAB      000005D8 RG 04
REQ_COMMAND    000001D7 RG 02
RETURN_NODES   000000DF R 05
RMSS_EOF       ***** X 05
SLPSB_FILENO   = 0000000D
SLPSB_FLAGS    = 0000000C
SLPSK_BLN      = 000000A8
SLPSL_BWD      = 00000004
SLPSL_FWD      = 00000000
SLPSQ_AUCDS    = 00000028
SLPSQ_AUDDS    = 00000010
SLPSQ_CMNT     = 00000040
SLPST_AUCST    = 00000030
SLPST_AUDST    = 00000018
SLPST_NAM      = 00000048
SLPSW_DOT      = 0000000E
SLPSW_LOC1     = 00000008
SLPSW_LOC2     = 0000000A
SLPM_AUDIT     = 00000002 G
SLPM_OPEN      = 00000001 G
SLPM_SUPPRESS  = 00000004 G
SLPV_SUPPRESS  = 00000002 G
SLP_AUDDS      00000281 RG 02
SLP_AUDST      00000289 RG 02
SLP_FLAGS      0000027F RG 02
SLP_SIZE       000002AF RG 02
SS$CONTINUE    ***** X 05
SS$RESIGNAL    ***** X 05
START          00000000 RG 05
STSSK_WARNING  = 00000000

```

```

STSSM_FAC NO    = 0FFF0000
STSS$SEVERITY   = 00000003
STSSV_FAC SP    = 0000000F
STSSV_SEVERITY  = 00000000
SYSSCONNECT     ***** GX 05
SYSSFAOL        ***** GX 05
SYSSOPEN        ***** GX 05
SYSSPUT         ***** GX 05
SYSSUNWIND      ***** GX 05
VIRT_ADDR       000002AB RG 02

```

SUM  
Pse

PSE

---  
\$AB  
SUM  
SUM

Pha

---  
Ini  
Com  
Pas  
Sym  
Pas  
Sym  
Pse  
Cro  
Ass

The  
210  
The  
211  
17

Mac

---  
\$2  
-\$2  
TOT

492

The

MAC



+-----+  
! Psect synopsis !  
+-----+

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 ( 0.)	00 ( 0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
\$AB\$\$	00000000 ( 0.)	01 ( 1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
\$DATA	00000321 ( 801.)	02 ( 2.)	NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC BYTE
\$RMSNAM	00000017 ( 23.)	03 ( 3.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE
RMS	0000081C ( 2076.)	04 ( 4.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE
\$CODE	0000028A ( 650.)	05 ( 5.)	NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC BYTE
ERRORS	00000289 ( 649.)	06 ( 6.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE

+-----+  
! Performance indicators !  
+-----+

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.06	00:00:00.51
Command processing	105	00:00:00.72	00:00:02.35
Pass 1	327	00:00:12.07	00:00:28.94
Symbol table sort	0	00:00:01.32	00:00:03.18
Pass 2	109	00:00:02.60	00:00:06.00
Symbol table output	23	00:00:00.18	00:00:00.51
Psect synopsis output	3	00:00:00.03	00:00:00.03
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	598	00:00:16.98	00:00:41.53

The working set limit was 1200 pages.

64317 bytes (126 pages) of virtual memory were used to buffer the intermediate code.

There were 50 pages of symbol table space allocated to hold 877 non-local and 41 local symbols.

541 source lines were read in Pass 1, producing 30 object records in Pass 2.

40 pages of virtual memory were used to define 33 macros.

+-----+  
! Macro library statistics !  
+-----+

Macro library name	Macros defined
_\$255\$DUA28:[SYS.OBJ]LIB.MLB;1	0
-\$255\$DUA28:[SYS.LIB]STARLET.MLB;2	27
TOTALS (all libraries)	27

1128 GETS were required to define 27 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:SUMMAIN/OBJ=OBJ\$:SUMMAIN MSRC\$:SUMCOM/UPDATE=(ENH\$:SUMCOM)+MSRC\$:SUMMAIN/UPDATE=(ENH\$:SUMMAIN)+EXECML\$/LIB



0369

AH-BT13A-SE  
 VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY